

# A modified large-scale structure-preserving doubling algorithm for a large-scale Riccati equation from transport theory \*

Pei-Chang Guo <sup>†</sup>

School of Mathematical Sciences, Peking University, Beijing 100871, Beijing, China

## Abstract

We consider the large scale nonsymmetric algebraic Riccati equation arising in transport theory, where the  $n \times n$  coefficient matrices  $B, C$  are symmetric and low-ranked and  $A, E$  are rank one updates of nonsingular diagonal matrices. By introducing a balancing strategy and setting appropriate initial matrices carefully, we can simplify the large-scale structure-preserving doubling algorithm (SDAJs) for this special equation. We give modified large-scale structure-preserving doubling algorithm, which can reduce the flop operations of original SDAJs by half. Numerical experiments illustrate the effectiveness of our method.

**Keywords:** large-scale nonsymmetric algebraic Riccati equation, large-scale structure-preserving doubling algorithm, balancing strategy, appropriate initial matrices, transport theory.

## 1 Introduction

In nuclear physics, we need to solve the nonsymmetric algebraic Riccati equation (NARE):

$$XCX - XE - AX + B = 0 \quad (1.1)$$

and its dual equation

$$YBY - YA - EY + C = 0, \quad (1.2)$$

where  $A, B, C, E \in \mathbb{R}^{n \times n}$  are matrices given by

$$A = \Delta - eq^T, B = ee^T, C = qq^T, E = D - qe^T, \quad (1.3)$$

with

$$\begin{cases} e &= (1, 1, \dots, 1)^T, \\ q &= (q_1, q_2, \dots, q_n)^T, \quad q_i = \frac{c_i}{2\omega_i}, \\ \Delta &= \text{diag}(\delta_1, \delta_2, \dots, \delta_n), \quad \delta_i = \frac{1}{c\omega_i(1+\alpha)}, \\ D &= \text{diag}(d_1, d_2, \dots, d_n), \quad d_i = \frac{1}{c\omega_i(1-\alpha)}, \end{cases} \quad (1.4)$$

and  $0 < c \leq 1$ ,  $0 \leq \alpha < 1$ ,  $0 < \omega_n < \dots < \omega_2 < \omega_1$ ,  $\sum_{i=1}^n c_i = 1$ ,  $c_i > 0$ ,  $i = 1, 2, \dots, n$ .

These matrices and vectors rely on the parameters  $c$ ,  $\alpha$ ,  $\omega_i$  and  $c_i$ . For the further physical meaning of these parameters, please see [13] and the references therein. In this paper, NARE (1.1) is always referred to the nonsymmetric algebraic Riccati equation (1.1) associated with the special structure given by (1.3) and (1.4). In applications from transport theory, the minimal nonnegative

\*This research is supported by the National Natural Science Foundation under grant 11301013.

<sup>†</sup>Corresponding author, e-mail: gpeichang@126.com

solution of NARE (1.1) is of interest. The numerical algorithms about this minimal nonnegative solution have been studied by many authors in the past decade, and various direct and iterative methods have been proposed. The minimal nonnegative solution is associated with the matrix

$$H = \begin{bmatrix} E & -C \\ B & -A \end{bmatrix}.$$

The solution of NARE (1.1) can be expressed in closed form in terms of the eigenvalues and eigenvectors of  $H$ ; see [14] and [17]. Available iteration algorithms are the Newton's method [3, 5, 6, 16, 10], the fixed-point iteration method [2, 4, 8, 14, 15] and the structure-preserving doubling algorithm [7, 9, 1, 18].

In [9] the authors introduced a structure-preserving transformation and developed a structure-preserving doubling algorithm (SDA) approximating quadratically to the minimal nonnegative solution of NARE(1.1) and its dual equation. For medium size problems without the special structure SDA is very effective. In [18], the authors propose a large-scale structure-preserving doubling algorithm (SDA\_ls) based on the sparse-plus-low-rank coefficient matrix structure, which has  $O(n)$  computational complexity and memory requirement per iteration and could be applied to large-scale NARE satisfying the sparse-like structure.

In this paper, we further utilize the special structure of coefficient matrices of NARE (1.1) and improve the large-scale structure-preserving doubling algorithm. There are four matrix sequences  $\{E_k\}$ ,  $\{F_k\}$ ,  $\{H_k\}$  and  $\{G_k\}$  when using SDA. After an appropriate balancing of the coefficient matrices, we design appropriate initial matrices of SDA\_ls, which can reduce the flop operations of SDA\_ls by half. Our modified SDA\_ls maintains the same quadratic convergence rate as original SDA\_ls and is more efficient on large-scale NARE (1.1). The main contribution of this paper is that we show how to reduce the flop operations of SDA\_ls by half through introducing a balancing strategy and setting the appropriate initial matrices for SDA\_ls. We prove our result theoretically.

The rest of this paper is organized as follows. In section 2 some preliminaries are presented. In section 3 we give the balancing strategy and the appropriate initial matrices for SDA\_ls, showing how to reduce the flop operations by half at each iteration step. In section 4 we present some numerical examples, which show that our modified SDA\_ls is much faster than original SDA\_ls applied to NARE (1.1). Throughout the paper, the Hadamard product of  $A$  and  $B$  is defined by  $A \circ B = (a_{ij}b_{ij})$ ,  $I$  is used to denote the identity matrix of appropriate dimension,  $\oplus$  is used to denote the direct sum of square matrices.

## 2 Preliminaries

### 2.1 Structure-Preserving Doubling Algorithm

The structure-preserving doubling algorithm [9] is quadratically convergent to computing the minimal positive solution of NARE(1.1). The algorithm can be described as follows.

Choose  $\gamma \geq \max \{e_{ii}, a_{ii} : i = 1, \dots, n\}$ ; let

$$W = A + \gamma I - B(E + \gamma I)^{-1}C, V = E + \gamma I - C(A + \gamma I)^{-1}B,$$

and

$$\begin{aligned} E_0 &= I - 2\gamma V^{-1}, & F_0 &= I - 2\gamma W^{-1}, \\ G_0 &= 2\gamma(E + \gamma I)^{-1}CW^{-1}, & H_0 &= 2\gamma W^{-1}B(E + \gamma I)^{-1}. \end{aligned} \quad (2.1)$$

where  $A, B, C, E$  are coefficient matrices of (1.1) and  $e_{ii}$  and  $a_{ii}$  are the  $i$ -th diagonal elements of the matrices  $E$  and  $A$ , respectively.

For  $k \geq 0$ , calculate

$$\begin{aligned} E_{k+1} &= E_k(I - G_k H_k)^{-1} E_k, \\ F_{k+1} &= F_k(I - H_k G_k)^{-1} F_k, \\ G_{k+1} &= G_k + E_k(I - G_k H_k)^{-1} G_k F_k, \\ H_{k+1} &= H_k + F_k(I - H_k G_k)^{-1} H_k E_k. \end{aligned} \quad (2.2)$$

Let

$$K = \begin{bmatrix} E & -C \\ -B & A \end{bmatrix}$$

be a nonsingular  $M$ -matrix. If  $\lambda_1, \lambda_2, \dots, \lambda_{2n}$  are the eigenvalues of  $H$  ordered by nonincreasing real parts, then the eigenvalues of  $K$  are  $\lambda_1, \dots, \lambda_n, -\lambda_{n+1}, \dots, -\lambda_{2n}$ [5].

From [7, 12], We have the following convergence result.

**Lemma 2.1.** *If the matrix sequences  $\{E_k\}, \{F_k\}, \{H_k\}$  and  $\{G_k\}$  are generated by (2.2), and  $K$  is nonsingular, then  $\lim_{k \rightarrow \infty} E_k = \lim_{k \rightarrow \infty} F_k = 0$ ,  $\lim_{k \rightarrow \infty} G_k = Y$ , and  $\lim_{k \rightarrow \infty} H_k = X$  with quadratic convergence rate, where  $X$  and  $Y$  are the minimal nonnegative solutions of NARE (1.1) and its dual equation (1.2), respectively.*

## 2.2 Large-scale SDA

To develop a large-scale SDA [18], The authors assume that  $E, A$  are sparselike (with the products  $A^{-1}u, A^{-T}u, E^{-1}u, E^{-T}u$  computable in  $O(n)$  complexity where  $u$  is a vector) and  $B, C \in \mathbb{R}^{n \times n}$  have the full low-ranked decompositions

$$B = B_1 B_2^T, C = C_1 C_2^T, \quad (2.3)$$

where  $B_1, C_1 \in \mathbb{R}^{n \times r}$  and  $B_2, C_2 \in \mathbb{R}^{r \times n}$  with  $r \ll n$ . The authors apply the Sherman-Morrison-Woodbury formula to avoid the invention of large matrices and use low ranked matrices to represent iterates. The basic large-scale SDA (SDA\_ls) is as follows.

For  $k = 1, 2, \dots$ , SDA\_ls is organized so that the iterates have the recursive forms

$$\begin{aligned} H_k^\tau &= Q_{1k}^\tau \Sigma_k^\tau Q_{2k}^{\tau T}, & G_k^\tau &= P_{1k}^\tau \Gamma_k^\tau P_{2k}^{\tau T}, \\ E_k^\tau &= E_{k-1}^{\tau 2} + E_{1k}^\tau E_{2k}^{\tau T}, & F_k^\tau &= F_{k-1}^{\tau 2} + F_{1k}^\tau F_{2k}^{\tau T}, \end{aligned} \quad (2.4)$$

where  $Q_{ik}^\tau$  and  $P_{ik}^\tau$  are column orthogonal with widths being  $m_k$  and  $l_k$  ( $i = 1, 2$ ) respectively. Note that  $E_k^\tau$  and  $F_k^\tau$  are not formed explicitly.

For the initial matrices (2.1), we have

$$E_0 = I - 2\gamma V^{-1}, F_0 = I - 2\gamma W^{-1}, H_0 = Q_{10} \Sigma_0 Q_{20}^T, G_0 = P_{10} \Gamma_0 P_{20}^T,$$

where

$$Q_{10} \equiv 2\gamma W^{-1} B_1, \quad Q_{20} \equiv (E + \gamma I)^{-T} B_2, \quad \Sigma_0 \equiv I, \quad (2.5)$$

$$P_{10} \equiv 2\gamma (E + \gamma I)^{-1} C_1, \quad P_{20} \equiv W^{-T} C_2, \quad \Gamma_0 \equiv I. \quad (2.6)$$

Note that efficient linear solvers with  $O(n)$  complexity for large scale  $A, E$  and  $A + \gamma I, E + \gamma I$  are available. Utilizing Shermann-Morrison-Woodbury (SMW) formula  $W^{-1}u$  and  $V^{-1}u$  can be computed with  $O(n)$  complexity too. Then we compute the economic QR decompositions

$$Q_{10} = \hat{Q}_{10} R_{1q}, Q_{20} = \hat{Q}_{20} R_{2q}, P_{10} = \hat{P}_{10} R_{1p}, P_{20} = \hat{P}_{20} R_{2p}, \quad (2.7)$$

where  $\hat{Q}_{10}, \hat{Q}_{20}, \hat{P}_{10}, \hat{P}_{20}$  are column orthogonal and  $R_{1q}, R_{2q}, R_{1p}, R_{2p}$  are upper triangular matrices with positive diagonal entries. Then we compute the SVDs

$$R_{1q} \Sigma_0 R_{2q}^T = [U_{10}^\tau, U_{10}^\epsilon] (\Sigma_0^\tau \oplus \Sigma_0^\epsilon) [U_{20}^\tau, U_{20}^\epsilon]^T, \|\Sigma_0^\epsilon\| < \epsilon_0,$$

$$R_{1p}\Gamma_0 R_{2p}^T = [V_{10}^\tau, V_{10}^\epsilon](\Gamma_0^\tau \oplus \Gamma_0^\epsilon)[V_{20}^\tau, V_{20}^\epsilon]^T, \|\Gamma_0^\epsilon\| < \epsilon_0, \quad (2.8)$$

where  $[U_{i0}^\tau, U_{i0}^\epsilon]$  and  $[V_{20}^\tau, V_{20}^\epsilon]$  ( $i = 1, 2$ ) are orthogonal,  $\Sigma_0^\tau \oplus \Sigma_0^\epsilon$  and  $\Gamma_0^\tau \oplus \Gamma_0^\epsilon$  are nonnegative diagonal. By setting

$$Q_{10}^\tau = \hat{Q}_{10}U_{10}^\tau, Q_{20}^\tau = \hat{Q}_{20}U_{20}^\tau, P_{10}^\tau = \hat{P}_{10}V_{10}^\tau, P_{20}^\tau = \hat{P}_{20}V_{20}^\tau, \quad (2.9)$$

we get the truncated initial matrices for large-scale SDA (SDA\_ls)

$$E_0^\tau = E_0, F_0^\tau = F_0, H_0^\tau = Q_{10}^\tau \Sigma_0^\tau Q_{20}^{\tau T}, G_0^\tau = P_{10}^\tau \Gamma_0^\tau P_{20}^{\tau T}. \quad (2.10)$$

Then we compute

$$\begin{aligned} \check{\Sigma}_k^\tau &= \Sigma_k^\tau + \Sigma_k^\tau Q_{2k}^{\tau T} P_{1k}^\tau \Gamma_k^\tau (I - P_{2k}^{\tau T} H_k^\tau P_{1k}^\tau \Gamma_k^\tau)^{-1} P_{2k}^{\tau T} Q_{1k}^\tau \Sigma_k^\tau, \\ \check{\Gamma}_k^\tau &= \Gamma_k^\tau + (I - \Gamma_k^\tau P_{2k}^{\tau T} H_k^\tau P_{1k}^\tau)^{-1} \Gamma_k^\tau P_{2k}^{\tau T} H_k^\tau P_{1k}^\tau \Gamma_k^\tau, \end{aligned} \quad (2.11)$$

and

$$\begin{aligned} F_{1,k+1}^\tau &= F_k^\tau Q_{1k}^\tau (I - \Sigma_k^\tau Q_{2k}^{\tau T} G_k^\tau Q_{1k}^\tau)^{-1} \Sigma_k^\tau Q_{2k}^{\tau T} P_{1k}^\tau \Gamma_k^\tau, & F_{2,k+1}^\tau &= F_k^{\tau T} P_{2k}^\tau, \\ E_{1,k+1}^\tau &= E_k^\tau P_{1k}^\tau (I - \Gamma_k^\tau P_{2k}^{\tau T} H_k^\tau P_{1k}^\tau)^{-1} \Gamma_k^\tau P_{2k}^{\tau T} Q_{1k}^\tau \Sigma_k^\tau, & E_{2,k+1}^\tau &= E_k^{\tau T} Q_{2k}^\tau. \end{aligned} \quad (2.12)$$

From the economic QR decompositions

$$[Q_{1k}^\tau, F_k^\tau Q_{1k}^\tau] = [Q_{1k}^\tau, \hat{Q}_{1k}] \begin{bmatrix} I & S_{1q} \\ 0 & R_{1q} \end{bmatrix}, [Q_{2k}^\tau, E_k^\tau Q_{2k}^\tau] = [Q_{2k}^\tau, \hat{Q}_{2k}] \begin{bmatrix} I & S_{2q} \\ 0 & R_{2q} \end{bmatrix}, \quad (2.13)$$

$$[P_{1k}^\tau, E_k^\tau P_{1k}^\tau] = [P_{1k}^\tau, \hat{P}_{1k}] \begin{bmatrix} I & S_{1p} \\ 0 & R_{1p} \end{bmatrix}, [P_{2k}^\tau, F_k^\tau P_{2k}^\tau] = [P_{2k}^\tau, \hat{P}_{2k}] \begin{bmatrix} I & S_{2p} \\ 0 & R_{2p} \end{bmatrix}, \quad (2.14)$$

we compute

$$\begin{aligned} \hat{\Sigma}_{k+1} &= \begin{bmatrix} I & S_{1q} \\ 0 & R_{1q} \end{bmatrix} \begin{bmatrix} \Sigma_k^\tau & 0 \\ 0 & \check{\Sigma}_k^\tau \end{bmatrix} \begin{bmatrix} I & S_{2q} \\ 0 & R_{2q} \end{bmatrix}^T \\ \hat{\Gamma}_{k+1} &= \begin{bmatrix} I & S_{1p} \\ 0 & R_{1p} \end{bmatrix} \begin{bmatrix} \Gamma_k^\tau & 0 \\ 0 & \check{\Gamma}_k^\tau \end{bmatrix} \begin{bmatrix} I & S_{2p} \\ 0 & R_{2p} \end{bmatrix}^T, \end{aligned} \quad (2.15)$$

then we compute the SVDs

$$\begin{aligned} \hat{\Sigma}_{k+1} &= [U_{1,k+1}^\tau \quad U_{1,k+1}^\epsilon] (\Sigma_{k+1}^\tau \oplus \Sigma_{k+1}^\epsilon) [U_{2,k+1}^\tau \quad U_{2,k+1}^\epsilon]^T \\ \hat{\Gamma}_{k+1} &= [V_{1,k+1}^\tau \quad V_{1,k+1}^\epsilon] (\Gamma_{k+1}^\tau \oplus \Gamma_{k+1}^\epsilon) [V_{2,k+1}^\tau \quad V_{2,k+1}^\epsilon]^T \end{aligned} \quad (2.16)$$

with  $\|\Sigma_{k+1}^\epsilon\| < \epsilon_{k+1}$  and  $\|\Gamma_{k+1}^\epsilon\| < \epsilon_{k+1}$ . The truncated matrices are

$$\begin{aligned} Q_{1,k+1}^\tau &= [Q_{1k}^\tau, \hat{Q}_{1k}] U_{1,k+1}^\tau, & Q_{2,k+1}^\tau &= [Q_{2k}^\tau, \hat{Q}_{2k}] U_{2,k+1}^\tau, \\ P_{1,k+1}^\tau &= [P_{1k}^\tau, \hat{P}_{1k}] V_{1,k+1}^\tau, & P_{2,k+1}^\tau &= [P_{2k}^\tau, \hat{P}_{2k}] V_{2,k+1}^\tau. \end{aligned} \quad (2.17)$$

We note that if QR decompositions and SVDs are not introduced to truncate  $Q_{ik}, P_{ik}$  ( $i = 1, 2$ ) in SDA\_ls, then SDA\_ls is mathematically equivalent to SDA (2.2) because (2.4) are derived from (2.2) by applying SMW formula:

$$\begin{aligned} (I - H_k^\tau G_k^\tau)^{-1} &= I + Q_{1k}^\tau (I - \Sigma_k^\tau Q_{2k}^{\tau T} G_k^\tau Q_{1k}^\tau)^{-1} \Sigma_k^\tau Q_{2k}^{\tau T} G_k^\tau, \\ (I - G_k^\tau H_k^\tau)^{-1} &= I + P_{1k}^\tau (I - \Gamma_k^\tau P_{2k}^{\tau T} H_k^\tau P_{1k}^\tau)^{-1} \Gamma_k^\tau P_{2k}^{\tau T} H_k^\tau. \end{aligned}$$

### 3 Balancing Strategy and Modified SDA\_ls

In this section we introduce a balancing strategy, with which the matrix  $\tilde{X}^T$  is the minimal positive solution of the dual equation if  $\tilde{X}$  is the minimal positive solution of the algebraic Riccati equation.

Because the entries of the vector  $q$  are positive, we may define

$$\Phi := \text{diag}(\sqrt{q_1}, \dots, \sqrt{q_n}), \phi := (\sqrt{q_1}, \dots, \sqrt{q_n})^T.$$

Let

$$\begin{aligned} \tilde{X} &= \Phi X \Phi, \\ \tilde{A} &= \Phi A \Phi^{-1} = \Delta - \phi \phi^T, \\ \tilde{B} &= \Phi B \Phi = \phi \phi^T, \\ \tilde{C} &= \Phi^{-1} C \Phi^{-1} = \phi \phi^T = \tilde{B}, \\ \tilde{E} &= \Phi^{-1} E \Phi = D - \phi \phi^T. \end{aligned}$$

Then the algebraic Riccati equation of (1.1) can be equivalently expressed as

$$\tilde{X} \tilde{C} \tilde{X} - \tilde{X} \tilde{E} - \tilde{A} \tilde{X} + \tilde{B} = 0. \quad (3.1)$$

Obviously,  $X$  is a solution of (1.1) if and only if  $\tilde{X} = \Phi X \Phi$  is a solution of (3.1).

Let

$$\begin{aligned} \tilde{K} &= \begin{bmatrix} \tilde{E} & -\tilde{C} \\ -\tilde{B} & \tilde{A} \end{bmatrix} \\ &= \begin{bmatrix} \Phi^{-1} & 0 \\ 0 & \Phi \end{bmatrix} K \begin{bmatrix} \Phi & 0 \\ 0 & \Phi^{-1} \end{bmatrix}. \end{aligned}$$

Then  $\tilde{K}$  is similar to  $K$ .

Choose  $\tilde{\gamma} \geq \max\{\tilde{e}_{ii}, \tilde{a}_{ii} : i = 1, \dots, n\}$ ; let

$$\tilde{W} = \tilde{A} + \tilde{\gamma}I - \tilde{B}(\tilde{E} + \tilde{\gamma}I)^{-1}\tilde{C}, \tilde{V} = \tilde{E} + \tilde{\gamma}I - \tilde{C}(\tilde{A} + \tilde{\gamma}I)^{-1}\tilde{B}$$

and

$$\begin{aligned} \tilde{E}_0 &= I - 2\tilde{\gamma}\tilde{V}^{-1}, & \tilde{F}_0 &= I - 2\tilde{\gamma}\tilde{W}^{-1}, \\ \tilde{G}_0 &= 2\tilde{\gamma}(\tilde{E} + \tilde{\gamma}I)^{-1}\tilde{C}\tilde{W}^{-1}, & \tilde{H}_0 &= 2\tilde{\gamma}\tilde{W}^{-1}\tilde{B}(\tilde{E} + \tilde{\gamma}I)^{-1}, \end{aligned} \quad (3.2)$$

where  $\tilde{A}, \tilde{B}, \tilde{C}$  and  $\tilde{E}$  are coefficient matrices of (3.1) and  $\tilde{e}_{ii}$  and  $\tilde{a}_{ii}$  are the  $i$ -th diagonal elements of the matrices  $\tilde{E}$  and  $\tilde{A}$ , respectively..

For  $k \geq 0$ , calculate

$$\begin{aligned} \tilde{E}_{k+1} &= \tilde{E}_k(I - \tilde{G}_k\tilde{H}_k)^{-1}\tilde{E}_k, \\ \tilde{F}_{k+1} &= \tilde{F}_k(I - \tilde{H}_k\tilde{G}_k)^{-1}\tilde{F}_k, \\ \tilde{G}_{k+1} &= \tilde{G}_k + \tilde{E}_k(I - \tilde{G}_k\tilde{H}_k)^{-1}\tilde{G}_k\tilde{F}_k, \\ \tilde{H}_{k+1} &= \tilde{H}_k + \tilde{F}_k(I - \tilde{H}_k\tilde{G}_k)^{-1}\tilde{H}_k\tilde{E}_k. \end{aligned} \quad (3.3)$$

For the initial matrices of SDA\_ls for NARE (3.1), if we set

$$E_0 = I - 2\tilde{\gamma}\tilde{V}^{-1}, F_0 = I - 2\tilde{\gamma}\tilde{W}^{-1}, H_0 = Q_{10}\Sigma_0Q_{20}^T, G_0 = P_{10}\Gamma_0P_{20}^T,$$

where

$$\begin{aligned} Q_{10} &\equiv \sqrt{2\tilde{\gamma}}\tilde{W}^{-1}\phi, & Q_{20} &\equiv \sqrt{2\tilde{\gamma}}(\tilde{E} + \tilde{\gamma}I)^{-T}\phi, & \Sigma_0 &\equiv I, \\ P_{10} &\equiv \sqrt{2\tilde{\gamma}}(\tilde{E} + \tilde{\gamma}I)^{-1}\phi, & P_{20} &\equiv \sqrt{2\tilde{\gamma}}\tilde{W}^{-T}\phi, & \Gamma_0 &\equiv I, \end{aligned}$$

then the flop operations of large scale structure preserving doubling algorithm can be reduced by half. We show this in Theorem 3.1.

**Theorem 3.1.** For the initial matrices of SDA\_ls applied to NARE (3.1), if

$$E_0 = I - 2\tilde{\gamma}\tilde{W}^{-1}, F_0 = I - 2\tilde{\gamma}\tilde{W}^{-1}, H_0 = Q_{10}\Sigma_0Q_{20}^T, G_0 = P_{10}\Gamma_0P_{20}^T,$$

where

$$Q_{10} \equiv \sqrt{2\tilde{\gamma}}\tilde{W}^{-1}\phi, \quad Q_{20} \equiv \sqrt{2\tilde{\gamma}}(\tilde{E} + \tilde{\gamma}I)^{-T}\phi, \quad \Sigma_0 \equiv I, \quad (3.4)$$

$$P_{10} \equiv \sqrt{2\tilde{\gamma}}(\tilde{E} + \tilde{\gamma}I)^{-1}\phi, \quad P_{20} \equiv \sqrt{2\tilde{\gamma}}\tilde{W}^{-T}\phi, \quad \Gamma_0 \equiv I, \quad (3.5)$$

then for SDA\_ls

$$Q_{1k}^\tau = P_{2k}^\tau, \quad Q_{2k}^\tau = P_{1k}^\tau, \quad \Sigma_k^\tau = \Gamma_k^\tau, \quad (3.6)$$

$$E_k^\tau = E_k^{\tau T}, \quad F_k^\tau = F_k^{\tau T}, \quad H_k^\tau = G_k^{\tau T}, \quad (3.7)$$

$$E_k^\tau P_{1k}^\tau = E_k^{\tau T} Q_{2k}^\tau, F_k^\tau Q_{1k}^\tau = F_k^{\tau T} P_{2k}^\tau \quad (3.8)$$

hold true for  $k = 1, 2, \dots$ .

*Proof.* We prove the result by mathematical induction.

After being balanced, Since  $\tilde{B} = \tilde{C}$ ,  $\tilde{A}^T = \tilde{A}$  and  $\tilde{E}^T = \tilde{E}$ , we have

$$\tilde{W}^T = \tilde{W}, \tilde{V}^T = \tilde{V},$$

and, thus,

$$E_0^T = E_0, F_0^T = F_0, H_0 = G_0^T.$$

From

$$Q_{10} \equiv \sqrt{2\tilde{\gamma}}\tilde{W}^{-1}\phi, \quad Q_{20} \equiv \sqrt{2\tilde{\gamma}}(\tilde{E} + \tilde{\gamma}I)^{-T}\phi, \quad \Sigma_0 \equiv I, \\ P_{10} \equiv \sqrt{2\tilde{\gamma}}(\tilde{E} + \tilde{\gamma}I)^{-1}\phi, \quad P_{20} \equiv \sqrt{2\tilde{\gamma}}\tilde{W}^{-T}\phi, \quad \Gamma_0 \equiv I,$$

we get

$$Q_{10} = P_{20}, Q_{20} = P_{10}, \Sigma_0 = \Gamma_0^T.$$

Then from the initial process of SDA\_ls, i.e. (2.7), (2.8), (2.9), (2.10), we can verify that

$$Q_{10}^\tau = P_{20}^\tau, \quad Q_{20}^\tau = P_{10}^\tau, \quad \Sigma_0^\tau = \Gamma_0^{\tau T}, \\ E_0^\tau = E_0^{\tau T}, \quad F_0^\tau = F_0^{\tau T}, \quad H_0^\tau = G_0^{\tau T},$$

and

$$E_0^\tau P_{10}^\tau = E_0^{\tau T} Q_{20}^\tau, F_0^\tau Q_{10}^\tau = F_0^{\tau T} P_{20}^\tau.$$

Hence the result is true for  $k = 0$ . Assume that the result holds true for  $k = l$ , and we consider the case of  $k = l + 1$ . Note that at each local step of SDA\_ls, the local truncation is done only on matrices  $Q_{ik}$  and  $P_{ik}$  ( $i=1,2$ ), so the following iteration processes of SDA\_ls

$$E_{l+1}^\tau = E_l^{\tau 2} + E_{1,l+1}^\tau E_{2,l+1}^{\tau T}, F_{l+1}^\tau = F_l^{\tau 2} + F_{1,l+1}^\tau F_{2,l+1}^{\tau T},$$

are mathematically equivalent to applying SMW formula to

$$E_{l+1}^\tau = E_l^\tau (I - G_l^\tau H_l^\tau)^{-1} E_l^\tau, F_{l+1}^\tau = F_l^\tau (I - H_l G_l^\tau)^{-1} F_l^\tau. \quad (3.9)$$

From the assumption of induction we know

$$E_l^\tau = E_l^{\tau T}, \quad F_l^\tau = F_l^{\tau T}, \quad H_l^\tau = G_l^{\tau T},$$

so from (3.9) we get

$$E_{l+1}^\tau = E_{l+1}^{\tau T}, F_{l+1}^\tau = F_{l+1}^{\tau T}.$$

By direct calculations we can verify that

$$\check{\Sigma}_l^\tau = \check{\Gamma}_l^{\tau T}$$

holds in the iteration step (2.11) and we have

$$[Q_{1l}^\tau, F_l^\tau Q_{1l}^\tau] = [P_{2l}^\tau, F_l^\tau P_{2l}^\tau], [Q_{2l}^\tau, E_l^\tau Q_{2l}^\tau] = [P_{1l}^\tau, E_l^\tau P_{1l}^\tau]$$

in iteration step (2.13) because of assumption of induction, so we get

$$\hat{\Sigma}_{l+1} = \hat{\Gamma}_{l+1}^T$$

in the iteration step (2.15). Then from (2.16) we get

$$\Sigma_{l+1}^\tau = \Gamma_{l+1}^{\tau T}$$

and from (2.17) we get

$$Q_{1,l+1}^\tau = P_{2,l+1}^\tau, Q_{2,l+1}^\tau = P_{1,l+1}^\tau,$$

and thus

$$H_{l+1}^\tau = G_{l+1}^{\tau T}$$

Combining with

$$E_{l+1}^\tau = E_{l+1}^{\tau T}, F_{l+1}^\tau = F_{l+1}^{\tau T},$$

we see that

$$E_{l+1}^\tau P_{1,l+1}^\tau = E_{l+1}^{\tau T} Q_{2,l+1}^\tau, F_{l+1}^\tau Q_{1,l+1}^\tau = F_{l+1}^{\tau T} P_{2,l+1}^\tau.$$

So we know that the conclusion holds true for  $k = l+1$ . By induction we have proved the result.  $\square$

Based on the above discussion, we present the modified SDA\_ls algorithm for NARE (1.1) as follows.

#### Modified SDA\_ls for NARE (1.1)

1. Set  $\phi := (\sqrt{q_1}, \dots, \sqrt{q_n})^T$  and let

$$A = \Delta - \phi\phi^T, B = \phi\phi^T, C = \phi\phi^T, E = D - \phi\phi^T.$$

2.  $k := 0$ , compute  $W^{-1}, V^{-1}$  and  $E_0, F_0$  implicitly and set

$$Q_{10} \equiv \sqrt{2\gamma}W^{-1}\phi, Q_{20} \equiv \sqrt{2\gamma}(E + \gamma)^{-T}\phi, \Sigma_0 \equiv I.$$

Compute economic QR decompositions

$$Q_{10} = \hat{Q}_{10}R_{1q}, Q_{20} = \hat{Q}_{20}R_{2q}$$

and compute the SVDs

$$R_{1q}\Sigma_0R_{2q}^T = [U_{10}^\tau, U_{10}^\epsilon](\Sigma_0^\tau \oplus \Sigma_0^\epsilon)[U_{20}^\tau, U_{20}^\epsilon]^T, \|\Sigma_0^\epsilon\| < \epsilon_0.$$

Set

$$\begin{aligned} Q_{10}^\tau &= \hat{Q}_{10}U_{10}^\tau, Q_{20}^\tau = \hat{Q}_{20}U_{20}^\tau, \\ E_0^\tau &= E_0, F_0^\tau = F_0, H_0^\tau = Q_{10}^\tau\Sigma_0^\tau Q_{20}^{\tau T}. \end{aligned}$$

Table 1: Flop operations for the kth iteration in SDA\_ls

Computation	Flops
$\tilde{\Sigma}_k^\tau, \tilde{\Gamma}_k^\tau$	$4m_k m_k n$
$E_k^\tau P_{1k}^\tau, E_k^{\tau T} Q_{2k}^\tau$	$2[2^k(c_\gamma + 4 \sum_{j=1}^k 2^{-i} m_j)] m_k n$
$F_k^\tau Q_{1k}^\tau, F_k^{\tau T} P_{2k}^\tau$	$2[2^k(c_\gamma + 4 \sum_{j=1}^k 2^{-i} m_j)] m_k n$
$F_{1,k+1}^\tau, E_{1,k+1}^\tau$	$4m_k m_k n$
Orthogonalize $F_k^\tau Q_{1k}^\tau, F_k^{\tau T} P_{2k}^\tau, E_k^\tau P_{1k}^\tau, E_k^{\tau T} Q_{2k}^\tau$	$2[6(m_k^2 + m_k^2) + 2m_k] n$
$Q_{i,k+1}^\tau, P_{i,k+1}^\tau (i = 1, 2)$	$16(m_k m_{k+1}) n$

3. For  $k \geq 0$ , calculate

$$\tilde{\Sigma}_k^\tau = \Sigma_k^\tau + \Sigma_k^\tau Q_{2k}^{\tau T} Q_{2k}^\tau \Gamma_k^\tau (I - Q_{1k}^{\tau T} H_k^\tau Q_{2k}^\tau \Gamma_k^\tau)^{-1} Q_{1k}^{\tau T} Q_{1k}^\tau \Sigma_k^\tau,$$

and

$$\begin{aligned} F_{2,k+1}^\tau &= F_k^\tau Q_{1k}^\tau, & F_{1,k+1}^\tau &= F_{2,k+1}^\tau (I - \Gamma_k^\tau Q_{2k}^{\tau T} H_k^{\tau T} Q_{1k}^\tau)^{-1} \Sigma_k^\tau Q_{2k}^{\tau T} P_{1k}^\tau \Gamma_k^\tau, \\ E_{2,k+1}^\tau &= E_k^\tau Q_{2k}^\tau, & E_{1,k+1}^\tau &= E_{2,k+1}^\tau (I - \Gamma_k^\tau Q_{1k}^{\tau T} H_k^\tau P_{1k}^\tau)^{-1} \Gamma_k^\tau Q_{1k}^{\tau T} Q_{1k}^\tau \Sigma_k^\tau. \end{aligned}$$

Compute economic QR decompositions

$$[Q_{1k}^\tau, F_k^\tau Q_{1k}^\tau] = [Q_{1k}^\tau, \hat{Q}_{1k}] \begin{bmatrix} I & S_{1q} \\ 0 & R_{1q} \end{bmatrix}, [Q_{2k}^\tau, E_k^\tau Q_{2k}^\tau] = [Q_{2k}^\tau, \hat{Q}_{2k}] \begin{bmatrix} I & S_{2q} \\ 0 & R_{2q} \end{bmatrix},$$

and

$$\hat{\Sigma}_{k+1} = \begin{bmatrix} I & S_{1q} \\ 0 & R_{1q} \end{bmatrix} \begin{bmatrix} \Sigma_k^\tau & 0 \\ 0 & \tilde{\Sigma}_k^\tau \end{bmatrix} \begin{bmatrix} I & S_{2q} \\ 0 & R_{2q} \end{bmatrix}^T.$$

Then compute the SVDs

$$\hat{\Sigma}_{k+1} = [U_{1,k+1}^\tau \quad U_{1,k+1}^\epsilon] (\Sigma_{k+1}^\tau \oplus \Sigma_{k+1}^\epsilon) [U_{2,k+1}^\tau, U_{2,k+1}^\epsilon]^T$$

with  $\|\Sigma_{k+1}^\epsilon\| < \epsilon_{k+1}$  and the truncated matrices

$$Q_{1,k+1}^\tau = [Q_{1k}^\tau, \hat{Q}_{1k}] U_{1,k+1}^\tau, Q_{2,k+1}^\tau = [Q_{2k}^\tau, \hat{Q}_{2k}] U_{2,k+1}^\tau,$$

and form

$$E_k^\tau = E_{k-1}^{\tau 2} + E_{1k}^\tau E_{2k}^{\tau T}, F_k^\tau = F_{k-1}^{\tau 2} + F_{1k}^\tau F_{2k}^{\tau T}$$

implicitly.

4. If the stopping criterion is satisfied, then return

$$X = (\Phi^{-1} Q_{1k}^\tau) \Sigma_k^\tau (Q_{2k}^{\tau T} \Phi^{-1}),$$

where  $\phi^{-1} = (\frac{1}{\sqrt{q_1}}, \dots, \frac{1}{\sqrt{q_n}})^T$ ; else go to step 3.

We compare the operation counts for the kth iteration in SDA\_ls and modified SDA\_ls. From Table 1 and Table 2, we see clearly the flops of modified SDA\_ls are roughly half of that of SDA\_ls.



Table 2: Flop operations for the kth iteration in modified SDA\_ls

Computation	Flops
$\tilde{\Sigma}_k^\tau$	$2m_k m_k n$
$E_k^\tau Q_{2k}^\tau$	$[2^k(c_\gamma + 4 \sum_{j=1}^k 2^{-j} m_j)] m_k n$
$F_k^\tau Q_{1k}^\tau$	$[2^k(c_\gamma + 4 \sum_{j=1}^k 2^{-j} m_j)] m_k n$
$F_{1,k+1}^\tau, E_{1,k+1}^\tau$	$4m_k m_k n$
Orthogonalize $F_k^\tau Q_{1k}^\tau, E_k^\tau Q_{2k}^\tau$	$[6(m_k^2 + m_k^2) + 2m_k] n$
$Q_{i,k+1}^\tau (i = 1, 2)$	$8(m_k m_{k+1}) n$

## 4 Numerical Experiments

## 5 Conclusions

We have presented a balancing strategy and specially chosen initial matrices for the large-scale structure-preserving doubling algorithm applied to NARE (1.1). We prove the flop operations of SDA\_ls at each iteration step can be reduced by half for this special equation. Computationally it is an interesting scheme that helps reducing flop operations and computing the solution faster. Numerical experiments show that modified SDA\_ls for NARE (1.1) is very effective and outperforms SDA\_ls.

## References

- [1] D. A. Bini, B. Meini and F. Poloni. Fast solution of a certain Riccati equation through Cauchy-like matrices. Electron. Trans. Numer. Anal., 33(1) (2008-2009) 84-104.
- [2] Z.-Z. Bai, Y.-H. Gao and L.-Z. Lu. Fast iterative Schemes for nonsymmetric algebraic Riccati equations arising from transport theory. SIAM J. Sci. Comput., 30 (2008) 804-818.
- [3] D. A. Bini, B. Iannazzo and F. Poloni. A fast Newtons method for a nonsymmetric algebraic Riccati equation. SIAM. J. Matrix Anal. Appl., 30 (2008) 276-290.
- [4] Z.-Z. Bai, X.-X. Guo and S.-F. Xu. Alternately linearized implicit iteration methods for the minimal nonnegative solutions of the nonsymmetric algebraic Riccati equations. Numer. Linear Algebra Appl., 13 (2006) 655-574.
- [5] C.-H. Guo. Nonsymmetric algebraic Riccati equations and Wiener-Hopf factorization for M-matrices. SIAM J. Matrix Anal. Appl., 23 (2001) 225-242.
- [6] C.-H. Guo and A. J. Laub. On the iterative solution of a class of nonsymmetric algebraic Riccati equations. SIAM J. Matrix Anal. Appl., 22 (2000) 376-391.
- [7] C.-H. Guo, B. Iannazzo and B. Meini. On the doubling algorithm for a (shifted) nonsymmetric algebraic Riccati equation. Technical Report TR 1628, Dipartimento di Matematica, University of Pisa, May 2006.
- [8] C.-H. Guo, W.-W. Lin. Convergence rates of some iterative methods for nonsymmetric algebraic Riccati equations arising in transport theory. Linear Algebra and its Applications, 432(1) (2010) 283-291.
- [9] X.-X. Guo, W.-W. Lin and S.-F. Xu. A structure-preserving doubling algorithm for nonsymmetric algebraic Riccati equation. Numer. Math., 103 (2006) 393-412.

- [10] X.-X. Guo and Z.-Z. Bai. On the minimal nonnegative solution of nonsymmetric algebraic Riccati equation. *J. Comput. Math.*, 23 (2005) 305-320.
- [11] P.-C. Guo and X.-X. Guo. A modified structure-preserving doubling algorithm for nonsymmetric algebraic Riccati equations from transport theory. *J. Comput. Appl. Math.*, 261 (2014) 213-220.
- [12] B. Iannazzo, F. Poloni. A subspace shift technique for nonsymmetric algebraic Riccati equations associated with an M-matrix. *Numer. Linear Algebra Appl.*, 20 (2013) 440-452.
- [13] J. Juang. Existence of algebraic matrix Riccati equations arising in transport theory. *Linear Algebra Appl.* 230 (1995) 89-100.
- [14] J. Juang and W.-W. Lin. Nonsymmetric algebraic Riccati equations and Hamiltonian-like matrices. *SIAM J. Matrix Anal. Appl.*, 20 (1998) 228-243.
- [15] L.-Z. Lu. Solution form and simple iteration of a nonsymmetric algebraic Riccati equation arising in transport theory. *SIAM J. Matrix Anal. Appl.*, 26 (2005) 679-685.
- [16] L.-Z. Lu. Newton iterations for a non-symmetric algebraic Riccati equation. *Numer. Linear Algebra Appl.*, 12 (2005) 191-200.
- [17] V. Mehrmann and H. Xu. Explicit solutions for a Riccati equation from transport theory. *SIAM J. Matrix Anal. Appl.*, 30 (2008) 1399-1357.
- [18] T.-X. Li, E. K. Chu, Y.-C. Kuo, and W.-W. Lin. Solving Large-Scale Nonsymmetric Algebraic Riccati Equations by Doubling. *SIAM J. Matrix Anal. Appl.*, 34 (2013) 1129-1147.